TITLE OF THE INVENTION

DEVICE AND METHOD FOR DOWNLOADING SOFTWARE VIA A NETWORK

5 BACKGROUND OF THE INVENTION

Field of the Invention

[0001]   The present invention relates to a device and method that use a network to download software.

Description of the Related Art

[0002]   Various types of payware adapted for use in personal computers (PCs) and game machines are known. The most common form of payware is the so-called "package sale", in which software written in media such as a floppy disk and a CD-ROM is sold. However, recently, with widespread use of the Internet, download sales have come into use, in which, by establishing a link to a network, software is downloaded from a server.

[0003]   The most ordinary method in which software is downloaded from a network is as follows:

A user searches for a site for software the user desires to use; the user operates a Web browser to connect to the Web server of a site found by the search; the user enters a credit card number into a displayed page in order to pay a charge, and subsequently initiates downloading to store the

software in a local terminal.

[0004]   The following software-downloading methods have also been proposed.

[0005]   For example, Japanese Patent Laid-Open No. 11-143719 discloses a public download system. In this system, a user loads a recording medium, which contains a download ID, into a download terminal linked to a server via a communication link, and data based on the ID is downloaded from the server and is written on the recording medium by the download terminal.

[0006]   According to Japanese Patent Laid-Open No. 09-288576, software is distributed beforehand to a user in a form in which it is stored on a recording medium protected by a secret code. A right code for enabling the software to be available for only a user-purchased time is distributed to a terminal for the user by computer communication.

[0007]   According to Japanese Patent Laid-Open No. 2000-092004, after a user who desires a service makes a contract with a broadcasting center to use software, a portable storage device that contains usable software information, including user-identification information given under the contract and a list of usable software names is sent from the broadcasting center to the user's home. Software transmitted from the broadcasting center is received and is used by the user's terminal. Accounting information in

accordance with how much the software is used is transmitted
to the broadcasting center, and the user is charged based on
the contract.

[0008]   Although package sales allow users to easily
acquire software without anxiety or special knowledge, there
is a problem in that, since the size of software increases
year after year, media for storing the software must have a
high capacity for storage.

[0009]   In a method in which software is distributed after
being stored on high-capacity media such as a CD-ROM, the
cost of the media itself is relatively inexpensive, but the
price of a reader device such as a CD-ROM drive is expensive.
Accordingly, the method has a problem in that the cost of
the reader device is high.

[0010]   In the case of storage media such as a Compact
Flash (CF) card, a reader device can be inexpensively made,
so that the cost of the reader device is low. Nevertheless,
the cost of unit capacity of CFs is high, which causes a
problem in that use of high-capacity software greatly
increases the cost of the media.

[0011]   An ordinary download-sale form has a problem in
that a user must perform complicated operations in which the
user searches a network for a desired download site, enters
the URL of the found site, and enters a credit card number.
The user may also feel anxious about entering the credit

card number.

[0012]   In the system of Japanese Patent Laid-Open No. 11-143719, the downloaded data is written on the recording medium containing the download ID. Thus, the recording medium, which is retained by the user, must have a large capacity matching the size of software to be downloaded. This causes a problem in that the required cost is high.

[0013]   In the case of Japanese Patent Laid-Open No. 09-288576, it is required that a reader device be capable of reading a medium such as a CD-ROM, which is distributed beforehand. A CD-ROM drive is more expensive than a card reader, etc. Accordingly, there is a problem similar to that of package sales in that the expensiveness increases the required cost.

[0014]   In the case of Japanese Patent Laid-Open No. 2000-092004, discussed above, it takes a lot of time to obtain software because the user must make a contract beforehand. In addition, since an IC card is sent, based on the contract from the broadcasting center, in a form that contains user identification information, it is purchased and carried by the user. Accordingly, there is a problem in that the user is unable to use the IC card immediately.

SUMMARY OF THE INVENTION

[0015]   In view of the foregoing problems, it is an object of the present invention to provide an information processing device and method, and a software distribution system in which software is distributed for sale both by using inexpensive media and by a facilitated user operation, while the cost of the device is minimized.

[0016]   It is another object of the present invention to provide a portable information storage medium used with the above device, method, and system.

[0017]   According to an aspect of the present invention, there is provided a portable information storage medium loadable into an information processing device connected to a network, which executes software that is downloaded from the network. The portable information storage medium includes a storage area for storing identification information on the software, location information representing a location on the network at which the software is stored, and secret information on a user who uses the software.

[0018]   According to another aspect of the present invention, there is provided an information processing device that includes a communication unit for communicating with a server terminal on a network, a portable-information-storage-medium connecting unit to which a portable information storage medium containing information on

software to be acquired via the network is connected, an information transfer unit for downloading the software from the server terminal into an internal storage medium by using the communication unit, a software storage unit for storing,

5   in a software storage area of the internal storage medium, the software downloaded into the internal storage medium, a software management unit for managing the software stored in the software storage area, and an external-storage-medium reading unit for reading predetermined information, which is

10 written on the portable information storage medium, when the portable information storage medium is connected to the portable-information-storage-medium connecting unit.

[0019]   According to another aspect of the present invention, there is provided a software distribution system

15 for distributing software provided by a software provider to a user via a store. The software provider includes a server terminal, which is externally accessed via a network so that software is downloaded, a user-information-management database connected to the server terminal, and a file server

20 terminal containing at least one type of software. The software provider supplies the store with a portable information storage medium containing a right of use of the software and information at the location on the network of the software. The store sells the portable information

25 storage medium to the user. The user downloads and executes

the software from the software provider after loading the purchased portable information storage medium into the information processing device.

[0020]   According to another aspect of the present invention, there is provided an information processing method including a communication step for communicating with a server terminal on a network, a portable-information-storage-medium connection step for connecting to a portable information storage medium containing information on software to be acquired via the network, an information transfer step for downloading the software from the server terminal into an internal storage medium, a software storage step for storing, in a software storage area of the internal storage medium, the software downloaded into the internal storage medium, a software management step for managing the software stored in the software storage area, and an external-storage-medium reading step for reading predetermined information, which is written on the portable information storage medium when the portable information storage medium is connected in the portable-information-storage-medium connecting step.

Further objects, features and advantages of the present invention will become apparent from the following description of the preferred embodiments with reference to the attached drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021]   Fig. 1 is a block diagram showing an information processing device according to a first embodiment of the present invention.

[0022]   Fig. 2 is a block diagram showing a structure of a software distribution system to which the first embodiment is applied.

[0023]   Fig. 3 is a flowchart showing a process performed when software is downloaded.

[0024]   Fig. 4 is a flowchart showing activation and termination of an application.

[0025]   Fig. 5 is a flowchart showing a process performed when an application is downloaded, in a case in which a process for checking a device ID is checked.

[0026]   Fig. 6 is a flowchart showing activation and termination of an application in a second embodiment of the present invention.

[0027]   Fig. 7 is a flowchart showing a process from loading of a CF card 202 up to mutual authentication in a third embodiment of the present invention.

[0028]   Fig. 8 is a flowchart showing a process from completion of mutual authentication up to activation of software in internal storage.

[0029]   Fig. 9 is a flowchart showing a process performed when a version of software in a server 301 is newer in a third embodiment of the present invention.

[0030]   Fig. 10 is a flowchart showing activation and termination of an application in a third embodiment of the present invention.

[0031]   Fig. 11 is a flowchart showing activation and termination of an application in a fourth embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

First Embodiment

[0032]   Fig. 1 shows the system structure of an information processing device 100 according to a first embodiment of the present invention. The information processing device 100 includes a system controller 1, a read-only memory (ROM) 2 for storing programs, a random access memory (RAM) 3 for temporarily storing information, an electrically erasable, programmable read-only memory (EEPROM) 4, an internal storage 5 for storing software and management information thereon, a user input interface 6, a mouse 7 and a keyboard 8, which are operated by a user, a picture output interface 9, a monitor 10, an external storage connection unit 11, and a local area network (LAN)

interface 12.

[0033]   The LAN interface 12 is connected to a LAN 14.   An external storage unit 13 is connected to the external storage connection unit 11.   The external storage unit 13 represents, for example, a Compact Flash (CF) card 202 (in Fig. 2) in the first embodiment.   Any type of readable and writable storage medium, such as a floppy disk or a SmartMedia, may be used as the external storage unit 13. The external storage connection unit 11 represents, for example, a CF-card reader.

[0034]   When main power is supplied, a program is loaded and executed by the system controller 1.   When the information processing device 100 is initially used by the user, a message instructing the user to input necessary user information is displayed on the monitor 10.   The user operates the keyboard 8 to input the user information.

[0035]   The inputted user information is received by the system controller 1 via the user input interface 6.   The system controller 1 stores the input user information in the internal storage 5.   The inputted user information includes a user name, an address, and a telephone number.   When the user finishes inputting the user information, the information processing device 100 is set in a normal operating state.

[0036]   The information processing device 100 not only

executes a program stored in the ROM 2, but also can read
and execute an application, which is stored beforehand in
the internal storage 5, when the user operates the mouse 7
to select a process from a menu displayed on the monitor 10.

[0037]   By also storing additional software in the
internal storage 5, a new application can be executed by the
information processing device 100. The process of storing
an additional application is as follows: the user purchases
an external storage unit 13 containing software at a store;
the user connects the external storage unit 13 to the
external storage connection unit 11; and by executing a
program to install the software, the software is installed
in the internal storage 5. After the installation, by
reading the software from the internal storage 5, the
software can be executed.

[0038]   In an ordinary application-sale method as
described above, the external storage unit 13 must have a
large capacity required for storing the entirety of software,
which increases the cost of the external storage unit 13.

[0039]   In the first embodiment, the external storage unit
13 has a small capacity and simply enables storage of the
identification information of the software and information
on the location of the software on a network. The software
itself can be downloaded from a server 301 (on the network),
which is indicated by the location information stored in the

external storage unit 13. The overall course summary of the
downloading is shown in Fig. 2.

[0040]   Fig. 2 shows a user's home 200, a software
provider 300, and a store 400 as main members.

[0041]   (1) The software provider 300 supplies the store
400 with a CF card 202 containing the software information.

[0042]   (2) A user 206 goes to the store 400 and purchases
the CF card 202 from a sales person 401. The CF card 202
contains manufacturer authentication information and
software information, as shown in Fig. 2.

[0043]   The CF card 202 is based on a known standard for
small memory cards. The CF card 202 has a length of 36.4 mm,
a width of 42.8 mm, and a thickness of 3.3 mm, and its
weight is, for example, approximately 14 grams.

[0044]   The CF card 202 contains at least the information
required to reach the software, such as a software name, a
product code, a version number, and a uniform resource
locator (URL) as the location on the network of the software,
and a serial number and a license key, which are information
indicating that the user 206 is an authorized purchaser.

[0045]   (3) The user 206 inserts the CF card 202 into a
slot 203 of an information processing device 201 at the
user's home 200.

[0046]   (4) The information processing device 201 writes
its device ID and user information in the CF card 202.

After that, the information processing device 201 uses a LAN 500 to access a server 301 of the software provider 300, which is indicated by the URL stored in the CF card 202.

[0047]   (5) The information processing device 201 downloads the software from the server 301.

[0048]   (6) The downloaded software is stored in internal storage of the information processing device 201.

[0049]   (7) The information processing device 201 writes a use record in the CF card 202. The information processing device 201 executes the software.

[0050]   (8) The user 206 unloads the CF card 202 after finishing the execution of the software.

[0051]   (9) At this time, the software is deleted from the internal storage. In order for the user 206 to use the software again, the user 206 inserts the CF card 202 again, and the processing from step (3) is executed.

[0052]   When the information processing device 201 writes the use record in step (7), it overwrites a number of times downloading is performed, which is written in the CF card 202, by a value in which one is added to the number.

[0053]   The overall course summary has been described. Next, with reference to Fig. 3, a process of the information processing device 201 from the insertion of the CF card 202 in step (3) to the deletion of the software from the internal storage 5 in step (9) is described below.

[0054]   When the user 206 inserts the CF card 202 into the slot 203, in step S101, the external storage connection unit 11 (in Fig. 1) detects the insertion of the CF card 202. In step S102, the information processing device 201 reads and stores the software information and the location information in the internal storage 205 (in Fig. 2).

[0055]   In step S103, the information processing device 201 determines whether or not user information set by the user 206 is written in the CF card 202. If the user information has not been written yet, the user information is read and is written in the CF card 202. In step S104, a device ID stored in the EEPROM (in Fig. 1) is written in the CF card 202, with the user information.

[0056]   The process proceeds to connection to the network. In step S105, based on the URL information read from the CF card 202, a connection request is sent to the LAN 14 (in Fig. 1) via the LAN interface 12. The URL information designates the server 301 (in Fig. 2) operated by the software provider 300 (in Fig. 2). The information processing device 201 then is connected to the server 301.

[0057]   In step S106, mutual authentication is performed by using manufacturer authentication information written in the CF card 202 and authentication information stored in the server 301. When mutual authentication is completed in step S107, the user information and the device ID are sent to the

server 301 (in Fig. 1).

[0058]   The server 301 stores the received information in
a user-management-information database 302 (in Fig. 2). The
server 301 may perform additional checking based on the
received information, if required. In this embodiment, the
server 301 checks nothing in particular, so the process
proceeds to step S109.

[0059]   In step S109, a product code, a serial number, and
a license key are sent as software information on
downloading to the server 301. The server 301 checks the
received information by querying a software-management-
information database 304. If the received information is
correct, software stored in a file server 303 is transmitted
to the user information device 201.

[0060]   In step S110, the user information device 201
receives the software from the server 301. In step S111,
the user information device 201 stores the software in the
internal storage 205 (in Fig. 2). After the downloading
ends, the process proceeds to an application activating
process in S112.

[0061]   Fig. 4 shows the flowchart of the application
activating process. In step S201, the software stored in
the internal storage 205 (in Fig. 2) is loaded into memory
and an application is activated. The user operates a mouse
to operate the user information device 201, and uses the

application while viewing a screen displayed on a monitor.

[0062]   When the user operates the mouse to select
termination of the application in step S202, the application
is terminated in step S203. After that, when the user
operates the mouse to select activation of the application
again in step S204, the process goes back to step S201 and
reactivates the application.

[0063]   When the user unloads the CF card 202 without
selecting the reactivation of the application, the unloading
of the CF card 202 is detected in step S205. In step S206,
the software (the application) is deleted, and the process
goes back for termination.

[0064]   In step S202, when termination of the application
is not selected, the application is being continuously used.
If the unloading of the CF card 202 is not detected, step
S202 and step S207 are repeatedly performed.

[0065]   When the user unloads the CF card 202 while the
application is being used, the unloading of the CF card 202
is detected in step S207. In step S208, after the execution
of the application is interrupted, a warning message, which
warns the user either to insert the CF card 202 or to
terminate the application, is displayed on the monitor, and
a user-input-awaiting state is activated.

[0066]   When the user inserts the CF card 202, the
insertion of the CF card 202 is detected in step S209. In

step S210, the warning message is deleted, and the execution
of the application is restarted in step S210. The process
goes back to step S202.

[0067]   When the user operates the mouse to select
termination of the application without inserting the CF card
202 after step S208, the insertion of the CF card 202 is not
detected in step S209, and in step S211, it is determined
whether or not termination of the application is selected.
If termination of the application is selected, the
application is terminated in step S212. In step S213, the
software (the application) is deleted from the internal
storage 205, and the process goes back for termination.

[0068]   The device ID written in the above step S104 can
be used for a security check function, as required. For
example, by comparing the device ID written in the CF card
202 is compared with the device ID stored in the EEPROM 4
when the CF card 202 is read, and only when both device IDs
match each other is it make possible to use the CF card 202.
A process into which this process is incorporated is shown
in Fig. 5.

[0069]   The process in Fig. 5 differs from the process in
Fig. 3 in a case in which the user information is written,
and proceeds to step S113. In step S113, the device ID
written in the CF card 202 is compared with the device ID
stored in the EEPROM 4 and when both match each other, the

server 301 designated by the URL is accessed via the network in step S105.

[0070]   When both device IDs do not match each other, in step S114, an error message is displayed and the process is terminated. The newly provided Step S113 is used to establish correspondence between the CF card 202 and each user's information processing device, whereby use of the CF card 202 in a different information processing device can be prevented.

Second Embodiment

[0071]   In the first embodiment, when the CF card 202 is unloaded while the application is being executed, the execution of the application is interrupted. However, in another technique, the application may be deleted after execution of the application is continued and is terminated. The flowchart of this case is shown in Fig. 6.

[0072]   In Fig. 6, step S701 is identical to the step S201 shown in Fig. 4, and steps S702 up to S706 are identical to steps S202 up to S206 shown in Fig. 4, respectively. Accordingly, a description of each step is omitted.

[0073]   When the user unloads the CF card 202 while the application is being executed, the unloading of the CF card 202 is detected in step S707, and in step S708, a flag indicating the unloading of the CF card 202 is set and the

application is continuously executed. When the user inserts
the CF card 202, the insertion of the CF card 202 is
detected in step S709. In step S710, the flag indicating
the unloading is deleted, and the process goes back to step
S702.

[0074]   When the user operates the mouse to select
termination of the application without inserting the CF card
202 after step S708, the insertion of the CF card 202 is not
detected in step S709, and the termination of the
application is selected by the user in step S711, so that in
step S712 the application is terminated. After that,
because the flag has been set, the application is deleted
from the internal storage 205 in step S713, and the process
goes back for termination. Processing up to the activation
of the application is identical to that in Fig. 3.

Third Embodiment

[0075]   In the first and second embodiments, when the CF
card 202 is unloaded from the information processing device
201, the application is deleted from the internal storage
205. However, by deleting only an option for executing the
application without deleting the application itself, the
user may be prevented from activating the application. This
is a third embodiment of the present invention, which is
described below.

[0076]   In the third embodiment, the physical structure of an information processing device is identical to the system structure shown in Fig. 1. Also, a process performed between the user, the store 400, and the software provider 300 is identical to that shown in Fig. 2.

[0077]   With reference to Figs. 7 to 10, a program downloading process of the third embodiment is described below.

[0078]   Fig. 7 shows a process from loading of the CF card 202, which is purchased by the user, into the information processing device 201 up to mutual authentication with the server 301. This process is almost identical to the process shown in Fig. 5 in the first embodiment, but differs only in that a case in which mutual authentication has been confirmed ("OK") is shown in Fig. 8. Since the other steps are identical to the corresponding steps shown in Fig. 5, a description of each step is omitted.

[0079]   The flowchart shown in Fig. 8 is described below.

[0080]   After the CF card 202 and the server 301 authenticate each other by using the manufacturer authentication information written in the CF card 202 and the authentication information in the server 301, the user information and the device ID are sent to the sever 301 in step S401. The server 301 stores the received information in the user-management-information database 302 (in Fig. 2).

The server 301 may perform additional checking based on the
received information, as required. In this embodiment, the
server 301 checks nothing in particular, so the process
proceeds to step S402.

5   [0081]   In step S402, a product code, a serial number, and
a license key are sent as software information on
downloading to the server 301. The server 301 checks the
received information by querying a software-management-
information database 304. If the received information is
10   correct, the process is continued. In step S403, the
process determines whether or not the software-information
storage area of the internal storage 205 (in Fig. 2)
includes software information in which a product number
matches a serial number. If software has already been
15   downloaded, the software-information storage area includes
the information and, in step S404, software-version
information is sent to the server 301. If the software has
never been downloaded, the software-information storage area
does not include the information and, in step S405, "version
20   zero" indicating that the software-information storage area
does not include the information is sent to the server 301.
[0082]   In step S406, the server 301 compares the received
version and the version of software stored in the server 301.
When the version of the software in the server 301 is not
25   newer, an instruction to use software in the internal

storage 205 is sent from the server 301. In step S407, the instruction is received by the information processing device 201. In step S408, an application activating process is performed in which the software in the internal storage 205 is activated. When the version of the software in the server 301 is newer, the process shown in Fig. 9 starts.

[0083]   Fig. 9 shows the rest of the downloading process performed when the version of the software in the server 301 is newer. The server 301 (in Fig. 2) transmits, to the information processing device 201, version information stored in the software-management-information database 304.

[0084]   In step S501, the version information is received from the server 301 and is stored in the software-information storage area by the information processing device 201. The server 301 transmits, to the information processing device 201, software stored in the file server 303. In step S502, the software is received by the information processing device 201. In step S503, the received software is stored in the downloaded software storage area 5 (in Fig. 1). After the downloading ends, an application activating process starts in step S504.

[0085]   Fig. 10 shows the application activating process. After the software stored in the internal storage 5 (in Fig. 1) is loaded into memory, an application of the software is activated in step S601. The user operates the mouse to

operate the application while viewing a screen displayed on the monitor.

[0086]   When the user operates the mouse to select termination of the application in step S602, in step S603, an option for activating the application is added to a menu screen displayed on the monitor.

[0087]   After that, when the user operates the mouse to select activation of the application from the displayed menu screen again, the process goes back to step S601, and the application is activated. When the user unloads the CF card 202 without selecting activation of the application again, the process determines, in step S605, whether or not unloading of the CF card 202 is detected. If unloading of the CF card 202 is detected, in step S606, the application activating option is deleted from the menu so that the application cannot be activated, and the process goes back for termination.

[0088]   When termination of the application is not selected in step S602, the application is in being continuously used. If unloading of the CF card 202 is not detected, steps S602 through step S607 are repeatedly performed.

[0089]   When the user unloads the CF card 202 while the application is being used, the unloading of the CF card 202 is detected in step S607. In step S608, after execution of

the application is interrupted, a warning message warning the user either to insert the CF card 202 or to terminate the application is displayed on the monitor, and a user-input-awaiting state is activated.

[0090]   When the user inserts the CF card 202, the insertion of the CF card 202 is detected in step S609. In step S610, the warning message is deleted, and execution of the application is restarted in step S610. The process goes back to step S602.

[0091]   When the user operates the mouse to select termination of the application without inserting the CF card 202 after step S608, insertion of the CF card 202 is not detected in step S609 and, in step S611, it is determined whether or not termination of the application is selected. If termination of the application is selected, the application is terminated in step S612. In step S613, the application activating option is deleted from the displayed menu, and the process goes back for termination.

Fourth Embodiment

[0092]   In the above third embodiment, when the CF card 202 is unloaded from the information processing device 201 while the application is being executed, execution of the application is interrupted. However, in another technique, the application activating option may be deleted after

execution of the application is continued without being
interrupted, and is terminated. The flowchart of this case
is shown in Fig. 11.

[0093]   With reference to Fig. 11, a fourth embodiment of
the present invention is described below.

[0094]   Step S801 in Fig. 11 is identical to step S601 in
Fig. 10, and steps S802 to S806 are identical to steps S602
to S606, respectively. Accordingly, a description of each
step is omitted.

[0095]   When the user unloads the CF card 202 while the
application is being executed, the unloading of the CF card
202 is detected in step S807. In step S808, a flag
indicating that the CF card 202 has been unloaded is set,
and execution of the application is continued. If the user
inserts the CF card 202, the insertion of the CF card 202 is
detected in step S809. In step S810, the flag is deleted
and execution of the application is continued. After that,
the process goes back to step S802.

[0096]   When the user operates the mouse to select
termination of the application without inserting the CF card
202 after step S808, insertion of the CF card 202 is not
detected in step S809 and, in S811, termination of the
application is selected by the user. In step S812, the
application is terminated.

[0097]   After that, because the flag has been set, the

application activating option is deleted from the displayed

menu in step S813, and the process goes back for termination.

In Fig. 11, steps up to the activation of the application is

are identical to those in Figs. 7 to 9.

5

Other Embodiments of the Present Invention

[0098]   Components as described in the foregoing

embodiments each include a central processing unit or a

microprocessor unit, a random access memory (RAM), and a

read-only memory (ROM) of a computer. Thus, the components

can be implemented such that a program stored in the RAM or

the ROM operates. Accordingly, by recording in a recording

medium, such as a CD-ROM, a program for controlling the

computer to perform the functions, and loading the program

into the computer, the components can be implemented. Not

only a CD-ROM, but also a floppy disk, a hard disk, a

magnetic tape, an magneto-optical disk, a non-volatile

memory card, and the like, can be used as the recording

medium.

[0099]   The embodiments of the present invention include,

in addition to a case in which the functions of the

foregoing embodiments are implemented such that a computer

executes a supplied program, a case in which the functions

of the foregoing embodiments are implemented such that the

program operates in cooperation with an operating system of

the computer or with other application software, and a case
in which the functions of the foregoing embodiments are
implemented such that all or part of a process based on the
supplied program is performed by an add-in board or an add-
in unit for the computer.

[0100]   In addition, all or part of the program may be
executed by another computer so that the present invention
can be used in network environments. For example, a screen
input process may be performed by a remote terminal computer,
and various types of determination, logging, etc., may be
performed by another computer such as a central computer.

[0101]   As described above, according to the present
invention, a right-to-use software can be sold by using a
small-capacity, low-cost, portable information storage
medium, and the software can be executed after being
automatically downloaded from a server terminal, without
requesting a user's operation.

[0102]   According to another feature of the present
invention, software is deleted when a portable information
storage medium is unloaded. This enables only a person who
possesses the portable information storage medium to execute
the software, and unauthorized execution of the software can
be prevented.

[0103]   According to another feature of the present
invention, by writing the device ID of each device into each

portable information storage medium, the correspondence
between the device and the software of the portable
information storage medium can be established. This can
prevent unauthorized use of the portable information storage

5   medium, even if it is lost or stolen.

[0104]   While the present invention has been described
with reference to what are presently considered to be the
preferred embodiments, it is to be understood that the
invention is not limited to the disclosed embodiments. On

10   the contrary, the invention is intended to cover various
modifications and equivalent arrangements included within
the spirit and scope of the appended claims. The scope of
the following claims is to be accorded the broadest
interpretation so as to encompass all such modifications and

15   equivalent structures and functions.